Computer Communications 34 (2011) 2057–2071

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

A directional data dissemination protocol for vehicular environments ☆

Ramon S. Schwartz [a,*], Rafael R. R. Barbosa [b], Nirvana Meratnia [a], Geert Heijenk [b], Hans Scholten [a]

[a] Pervasive Systems (PS), Faculty of Electrical Engineering, Mathematics and Computer Science (EWI), University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands
[b] Design and Analysis of Communication Systems (DACS), Faculty of Electrical Engineering, Mathematics and Computer Science (EWI), University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands

ARTICLE INFO

Article history:
Received 7 July 2010
Received in revised form 9 December 2010
Accepted 15 March 2011
Available online 22 March 2011

Keywords:
Ad-Hoc Networks
VANET
Opportunistic network
Vehicular sensor network (VSN)
Dissemination Protocol

ABSTRACT

This paper presents a simple and robust dissemination protocol that efficiently deals with data dissemination in both dense and sparse vehicular networks. Our goal is to address highway scenarios where vehicles equipped with sensors detect an event, e.g., a hazard and broadcast an event message to a specific direction of interest. In order to deal with broadcast communication under diverse network densities, we design a dissemination protocol in such a way that: (i) it prevents the so-called broadcast storm problem in dense networks by employing an optimized broadcast suppression technique; and (ii) it efficiently deals with disconnected networks by relying on the store-carry-forward communication model. The novelty of the protocol lies in its simplicity and robustness. Simplicity is achieved by only considering two states (i.e., cluster *tail* and *non-tail*) for vehicles. Furthermore, vehicles in both directions help disseminating messages in a seamlessly manner, without resorting to different operation modes for each direction. Robustness is achieved by assigning message delivery responsibility to multiple vehicles in sparse networks. Our simulation results show that our protocol achieves higher delivery ratio and higher robustness when compared with DV-CAST under diverse road scenarios.

1. Introduction

Vehicular Ad-hoc Networks (VANETs) have gained considerable attention in the past few years due to their promising applicability with regard to safety, transport efficiency, and information/entertainment [1]. In particular, vehicular networks enable the opportunistic sensing of road environments that range from traffic flow and pollution monitoring to safety warning services [2]. Because modern vehicles are equipped with powerful processing units such networks, referred to as vehicular sensor networks (VSNs), are not typically affected by energy constraints and, therefore, differ considerably from traditional wireless sensor networks (WSNs).

For many of these applications, the data acquired by sensors, e.g., crash detection data, must be broadcast (disseminated) to all vehicles nearby. Because these events might not directly affect all vehicles within the event perimeter, broadcast messages can be propagated towards a specific direction such as to vehicles that are in fact approaching the dangerous area. In this paper, we consider the problem of coordinating these broadcast messages to a specific direction in a reliably, timely, and efficiently manner using vehicle-to-vehicle communications. We present a dissemination protocol which assumes no information available about the road topology. Therefore, in this work we focus on highway scenarios, where simple long bidirectional roads are present. For more complex topologies, such as the ones found in urban environments, geographical mapping information can be used to enhance our approach in future work.

In order to deal with broadcast communication, different dissemination strategies should be defined according to the current network situation. In dense networks, the number of broadcasts must be minimized to avoid excessive redundancy, contention, and collision rates [3]. These undesired factors collectively are referred to as the *broadcast storm problem*. The minimization of these factors can be achieved by means of *broadcast suppression techniques* [4].

In sparse networks, on the other hand, the *store-carry-forward* communication model can take advantage of the mobility of nodes to store and transfer messages when nodes are geographically separated. This approach is commonly put in practice in delay tolerant/opportunistic networks [5,6]. In this type of network, vehicles assess the best available opportunities of connectivity for data dissemination and make decisions solely based on current knowledge. For instance, at the moment of an encounter of

☆ This work is supported by iLAND Project, ARTEMIS Joint Undertaking Call for proposals ARTEMIS-2008-1, Project contract no.100026; and by the Dutch Senter Novem/HTAS (High Tech Automotive Systems) Project Connect & Drive, Project no. HTASD08002.

* Corresponding author.
E-mail addresses: r.s.schwartz@utwente.nl, r.desouzaschwartz@utwente.nl (R.S. Schwartz), r.barbosa@utwente.nl (R.R. R. Barbosa), n.meratnia@utwente.nl (N. Meratnia), geert.heijenk@utwente.nl (G. Heijenk), hans.scholten@utwente.nl (H. Scholten).

0140-3664/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2011.03.007

multiple vehicles, each vehicle carrying data evaluates the probability that other vehicles would lead this data to its final destination or geographical region of interest. For this assessment, available knowledge such as the vehicle's direction or even the complete route of a vehicle set in a navigation system is taken in account. In this paper, we concentrate on exploiting the movement pattern of vehicles in one direction to cope with separate vehicle clusters in the other direction as motivated in [7,8].

The contribution of this work lies in combining an optimized broadcast suppression technique with the store-carry-forward model in a single dissemination protocol called the simple and robust dissemination (SRD) protocol. Such protocol operates seamlessly in the network layer in both dense and sparse networks. This paper extends our previous work published in [9] with a much richer detailed explanation of the protocol and with a complete performance evaluation which includes several new aspects such as the direct comparison with DV-CAST [10]; up to best of our knowledge, the only protocol that also focus on directional broadcasting in both dense and sparse highways. By means of simulations under diverse highway scenarios, we show that SRD presents better overall performance in terms of delivery ratio and robustness when compared with DV-CAST.

The remainder of this paper is organized as follows. Section 2 provides an overview and brief comparison between SRD and other proposals with respect to suppression techniques and protocols that deal with sparse networks. Next, Section 3 describes the SRD protocol in detail. Section 4 describes the performance evaluation of the protocol carried out by means of simulations. Finally, Section 5 concludes this paper and outlines our future directions.

2. Related work

Various solutions for VANETs have been proposed to cope with message dissemination under different traffic conditions. In dense scenarios, *suppression* techniques have been proposed to address the so-called broadcast storm problem. For a given broadcast message, the solution of this problem consists in finding the minimum set of nodes capable of reaching all other nodes in the network. If only nodes in this set broadcast this message, redundancy is kept to a minimum. In the context of mobile Ad-Hoc networks (MANETs) several solutions to address this problem are proposed, e.g., [11]. However, these solutions are generally not optimal for VANET scenarios, due to basic dissimilarities with MANETs. First, the mobility of nodes in VANETs is constrained to roads, so it is possible to employ simpler methods to select nodes that should broadcast a given message. In addition, solutions for VANETs should cope with much higher speeds when compared to MANETs.

Unlike MANETs, only a few suppression techniques have been proposed specifically for VANETs. In [4], three broadcast suppression techniques are presented to be employed in the network layer. Among these three techniques, Slotted 1-Persistence has achieved the best performance in terms of reducing the number of unnecessary broadcasts while still achieving a low end-to-end delay and high delivery ratio. This technique is time-based and non-probabilistic. Given a fixed number of time slots, the most distant vehicles in the message direction from the source vehicle, i.e., from where the message has been originated, will be given the earliest time slot to rebroadcast. Vehicles assigned to other time slots would then have time to cancel their transmissions upon the receipt of an echo. This would be an indication that the information has already been disseminated and redundant rebroadcasts are *suppressed*.

However, the Slotted 1-Persistence technique suffers from a synchronization problem [12,13] that can occur when multiple vehicles are assigned to a single time slot and start their transmissions simultaneously. This results in a substantial deterioration with respect to delivery ratio due to a higher number of collisions. In this paper, we tackle this problem by proposing the *Optimized Slotted 1-Persistence* technique (described in Section 3.3).

The use of vehicles moving in the opposite direction to help in message dissemination for sparse networks has been previously studied in [14,15,7,16,17]. In [14], three scenarios are considered: vehicles moving in the same or the opposite direction of the originator of the message and vehicles moving in both directions. Simulation results demonstrate that the use of vehicles moving in opposite directions improves the dissemination performance in many different scenarios. The directional propagation protocol [15] allows directional propagation of messages from a given point of origin. It requires the adoption of a cluster creation/maintenance mechanism and differentiates between inter and intra cluster communication. Further evaluation of this protocol in [7] has shown that vehicle mobility can be used to improve message propagation in scenarios in which conventional MANET protocols would fail due to the lack of end-to-end connectivity. The abiding geocast, described in [17], disseminates accident or congestion information to every vehicle passing through a warning zone during the event lifetime.

There is extensive work in the area of delay tolerant networks (DTN) and opportunistic networks for communication in disconnected networks. The *Epidemic routing* [18] uses flooding to disseminate messages through the network. In this approach nodes exchange data as soon as new neighbors are discovered. The *spray routing* [19] generates only a small number of message copies in order to ensure that the number of transmissions are small and controlled. In [20], moving relay nodes, referred to as *data mules*, are used to carry the messages to their destination. In the context of pocket switched networks (PSNs), where the nodes are devices carried by people, the *BUBBLE* algorithm is proposed [21]. It takes into account people's social relationships to select the nodes that can best relay messages. Once again, these approaches were designed assuming a different mobility model than the one present in VANETs, where the mobility of vehicles is constrained to single or multiple roads and usually follow well-defined rules, and for this reason they may not be suitable in this context. To the best of our knowledge, only two works apply the *store-carry-forward* mechanism for message dissemination in VANETs: the Distributed Vehicular Broadcast (DV-CAST) protocol [10] and the acknowledged parameterless broadcast in static to highly mobile (ackPBSM) [22].

The goal of the DV-CAST protocol is to adapt to different traffic densities, e.g., light traffic, moderate traffic, or traffic jam, while introducing a low overhead in high density situations and managing communication gaps in low density situations. Unlike in our protocol, DV-CAST relies on the periodic exchange of *hello messages* between all communicating vehicles. Especially in dense and dynamic networks, if not coordinated properly, hello messages might increase collision and contention, thus wasting bandwidth. Although our protocol also requires the exchange of periodic messages, a suppression technique is employed to prevent the so-called broadcast storm problem and to reduce the number of broadcasts. Our protocol also avoids the dependency on a single vehicle when bridging radio gaps in the network. All vehicles in the range of the vehicle positioned at the tail of a cluster act as *backup vehicles*, thus increasing robustness. Moreover, in [22] the DV-CAST protocol is reported to have a low reliability. This can be partially explained by unforeseen situations such as overtaking while determining the current traffic density. As in our protocol, a vehicle simply needs to determine whether it is the tail in a message direction, it does not suffer from this problem.

The ackPBSM protocol relies on the use of connected dominating Sets (CDS) to perform the broadcast. Similarly to DV-CAST, ackPBSM depends on the exchange of periodic hello messages to gather 1-hop position information. Acknowledge information is included in these messages in order to increase the delivery ratio and reduce transmission redundancy. The protocol is designed to operate in both highway and urban scenarios. A main difference when comparing to our protocol is that in ackPBSM all vehicles in the network are intended recipients of a broadcast. We argue that in many vehicular scenarios, a *directional* broadcast is more suitable. For instance, consider a highway with two directions. If an accident occurs in one direction, this information is only relevant to vehicles moving in that direction that have not yet reached that location. For vehicles that have already passed this location or are moving in the other direction, this information is irrelevant. Due to this difference, it is hard to quantitatively compare our protocol with ackPBSM. Therefore, we do not consider it in our performance evaluation presented in Section 4.

3. Protocol description

Our simple and robust dissemination (SRD) protocol aims to efficiently disseminate data in both dense and sparse vehicular networks. More specifically, it aims to achieve a high delivery ratio with a low propagation delay and yet without introducing an excessive load in the network. For this purpose, we take the following approach:

- In dense networks, we rely on an *optimized broadcast suppression technique* in order to cope with the *broadcast storm problem* by relaying messages using a low number of vehicles.
- In sparse networks, the *store-carry-forward* communication model is employed to deliver messages whenever a multi-hop connectivity among vehicles is not available.

3.1. Concept definitions

To better understand the protocol, we define the following concepts which are used throughout the remaining sections:

Definition 3.1 (*Message direction*). Given a multiple-lane road, where vehicles move in both road directions, the message direction d is defined by the application responsible for generating the message and it is one of the two road directions. We assume that this application can be running in either a vehicle or in a road-side unit (RSU), e.g., to broadcast a vehicle crash or an upcoming danger area. For the sake of simplicity, we refer to each direction as the easterly and westerly directions.

Definition 3.2 (*Vehicle cluster*). Given a multiple-lane road, where vehicles move in both easterly and westerly directions, a vehicle cluster vc is defined as a group of vehicles moving in any direction with multi-hop radio connectivity at a time instant.

Definition 3.3 (*Cluster front*). Given a vehicle cluster vc and a message direction d, the cluster front cf is defined as the vehicle within cluster vc with no radio connectivity with other vehicles positioned farther in the direction opposite to d. For instance, for an easterly message direction the front vehicle would be the farthest vehicle in the westerly direction within vc.

Definition 3.4 (*Cluster tail*). Given a vehicle cluster vc and a message direction d, the cluster tail ct is defined as the vehicle within cluster vc with no radio connectivity with other vehicles positioned farther in message direction d, i.e., the final vehicle belonging to vc in the message direction.

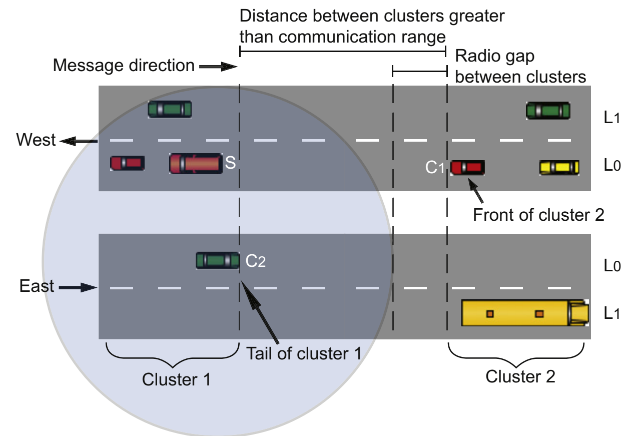


Fig. 1. Protocol concepts applied in a simple example.

Definition 3.5 (*Radio gap*). Given two vehicle clusters vc_1 and vc_2, message direction d, and cluster tail ct_1 of cluster vc_1, the gap is defined as:

$$Gap = D(vc_1, vc_2) - CR(ct_1) \quad (1)$$

where $D(vc_1, vc_2)$ is the distance between clusters vc_1 and vc_2, i.e., the distance between ct_1 and the cluster front of vc_2 in the message direction d. $CR(ct_1)$ is denoted as the communication range of vehicle ct_1.

Fig. 1 shows an example of how these concepts are applied. A road with multiple lanes contains vehicles moving in both westerly and easterly directions. Two groups of vehicles are separated by a radio gap and are classified as vehicle clusters vc_1 and vc_2. Let us suppose that a message is generated in vehicle S towards the easterly direction. With respect to this direction, vehicle C_2 is defined as the tail of vc_1 and C_1 as the front vehicle of vc_2.

3.2. Requirements and assumptions

For a proper operation of the SRD protocol, we require that vehicles are able to determine their position on the road using, for example, the global positioning system (GPS). It is not necessary that every vehicle is equipped with wireless communication devices. If they are, however, it is required that the radio ranges provided by these devices are symmetric and working at the same radio frequency, i.e., if a vehicle C_1 can communicate with a vehicle C_2, a transmission from C_2 will also reach C_1.

In this work, we assume that no roadside infrastructure is available. Although devices alongside the road could help broadcasting traffic events and possibly bridging radio gaps, we consider a highway scenario where only vehicles can generate and disseminate data. To this end, we assume that these vehicles are equipped with radio devices which comply with the IEEE 802.11p standard, the upcoming IEEE standard for the network PHY and MAC layers defined in [23].

3.3. Optimized slotted1-persistence

The suppression technique we propose in this work is based on the Slotted 1-Persistence presented in [4]; however, with a slightly altered formula to guarantee an equal distribution of vehicles among the time slots assigned[1]. The differences lie in two optimizations designed to decrease the number of collisions during a

[1] A typographical error with regard to the ceiling math function position has been identified in the formula for the Slotted 1-Persistence technique proposed in [4], which leads to inaccurate distribution of vehicles among different time slots.

rebroadcast and improve the overall delivery ratio: (i) assignment of different time slot priorities for each road direction; and (ii) the introduction of an additional delay within each time slot to cope with the synchronization problem found in [12,13]. We refer to our suppression mechanism as *Optimized Slotted 1-Persistence*.

The time slot assignment is defined as follows. If a vehicle j farther in the message direction receives a new message from vehicle i, it schedules a rebroadcast for that message; otherwise it ignores it and if the sender is farther in message direction it suppresses any previously scheduled rebroadcast for that message.

When scheduling a message, vehicle j first calculates the percentage distance PD_{ij} between the two vehicles with respect to the estimated transmission range R.

$$PD_{ij} = \left[\frac{min(D_{ij}, R)}{R}\right] \tag{2}$$

where D_{ij} is the distance between vehicles i and j. As a result, the PD_{ij} value will vary within the interval $(0,1]$ with large distances being closer to 1. The *minimum* function is necessary, since the transmission range R is an estimate based on the power level employed and vehicles in reality could be positioned at farther positions.

The time slot number S_{ij} assigned to vehicle j is then defined by the following equation:

$$S_{ij} = \lfloor NS \times (1 - PD_{ij}) \rfloor \tag{3}$$

where NS is the total number of time slots utilized. If vehicles are uniformly distributed within the transmission range of vehicle i, they will be equally distributed among the NS time slots reserved. S_{ij} will vary within the interval $[0, NS-1]$.

In most vehicular applications, a message should be forwarded only in one direction and the intended receivers are vehicles moving in one particular road direction. Therefore, the first (i) optimization we propose is to give a higher priority to one road direction and assign later time slots to vehicles in the opposite direction. The objective of this modification is to have fewer vehicles assigned to each time slot thereby reducing the number of rebroadcasts and collisions. We update the assignment of time slots S_{ij} to include this modification as follows:

$$S_{ij} = \begin{cases} \lfloor NS \times (1 - PD_{ij}) \rfloor & \textbf{if } v_{dir} = hp_{dir}; \\ \lfloor NS \times (2 - PD_{ij}) \rfloor & \textbf{if } v_{dir} \neq hp_{dir}. \end{cases} \tag{4}$$

where v_{dir} and hp_{dir} are the vehicle and high priority directions, respectively. In this way, the time slot range is equally divided in $[0, NS-1]$ for the high priority direction and $[NS, (2 \times NS) - 1]$ for the opposite (low priority) direction.

Finally, the time that vehicles have to wait before rebroadcasting at time slot S_{ij} is calculated by Eq. 5:

$$T_{Sij} = S_{ij} \times st \tag{5}$$

where the slot time st is a value larger than the one-hop delay that includes the medium access delay, transmission delay and propagation delay.

Assigning vehicles to different time slots clearly breaks the synchronization present in the simple flooding approach, where all nodes would rebroadcast simultaneously upon the receipt of a message. The slot time st is defined in such a manner that it gives vehicles assigned to later time slots the opportunity to cancel their transmissions, since the message has already been rebroadcast. Therefore, ideally only vehicles assigned to the earliest time slot would rebroadcast. However, a similar synchronization on a smaller scale can still occur when multiple vehicles are assigned to a single time slot and start their transmission simultaneously. Such a synchronization problem has been identified in [12]. To cope with this problem, in that work a variation of the Slotted 1-Persistence technique called microslotted 1-Persistence Flooding has been proposed. The proposed scheme functions in the same way as the Slotted 1-Persistence broadcasting scheme but with a small additional delay, i.e., the *micro slots*, within each time slot to break the defined synchronization. The same problem has been identified and referred to as the *timeslot boundary synchronization problem* in [13]. Differently, such work describes design guidelines for extra measures to be taken not only in the network layer but also in the link layer by inserting a pseudo-random delay to SIFS in the IEEE 802.11p MAC layer. Especially in congested networks, an additional delay introduced uniquely in the network layer does not suffice when nodes experience high contention in the link layer, as their timeslots could be again aligned.

As in [13], we support the position that the synchronization must be broken in both the network and link layers to be completely effective. However, in order to comply with the current MAC and PHY layers of the 802.11p standard, we propose the use of a small extra delay but only in networks layer and maintain the MAC layer unaltered. We follow the guidelines proposed in [13] that suggest that this extra delay must be chosen from a near continuous interval in order to completely break the alignment of timeslot boundaries. The additional delay AD_{ij} is our second (ii) optimization and is defined as follows:

$$AD_{ij} = \begin{cases} D_{max} \times (1 - PD_{ij}) & \textbf{if } v_{dir} = hp_{dir}; \\ D_{max} \times (2 - PD_{ij}) & \textbf{if } v_{dir} \neq hp_{dir}. \end{cases} \tag{6}$$

where D_{max} is the maximum delay allowed. Following the idea adopted for the assignment of time slots, vehicles driving in the high priority direction receive smaller delay values than vehicles driving in the opposite (low priority) direction.

The time that vehicles have to wait before rebroadcasting is updated to include the additional delay described and is expressed as:

$$T_{Sij} = (S_{ij} \times st) + AD_{ij} \tag{7}$$

The result of Eq. 7 is that for each road direction each time slot is stretched with an equal fraction of D_{max}. Moreover, the beginning of each time slot is shifted by the accumulated additional time of earlier time slots, thereby preserving the pre-defined st value and preventing overlapping between different time slots.

The complete suppression mechanism described in this section is shown in Fig. 2. In addition to the time slot assignment, the rule for canceling a rebroadcast also differs from the original Slotted 1-

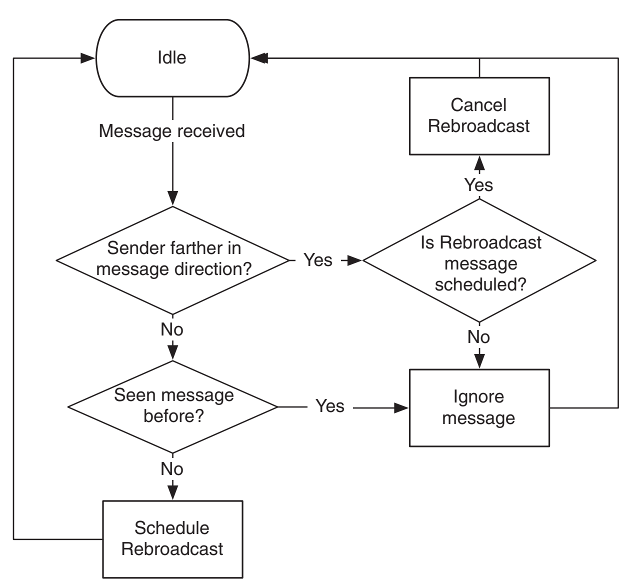


Fig. 2. The Optimized Slotted 1-Persistence technique.

Persistence. In [4], a suppression of a rebroadcast occurs whenever a duplicate of a scheduled message is received before its transmission time. Because the most distant vehicles in the message direction will rebroadcast first, there is no harm in unconditionally canceling rebroadcast when receiving an echo. However, due to our separation of priorities for each direction, the most distant vehicles can be positioned in the low priority direction and thus assigned to late time slots. As a consequence, these vehicles might cancel their rebroadcasts erroneously upon the receipt of an echo coming from vehicles closer to the sender (in the high priority direction) and impede a further propagation of the message. To prevent this behavior, we define that vehicles can only cancel their rebroadcasts when receiving an echo from other vehicles farther in the message direction. To guarantee that we only consider new messages, we keep track of the most recent message IDs in a list, namely, the last *m* message IDs received. New messages are managed by the *schedule rebroadcast* function which calculates the proper waiting time T_{Sij} and places the message in a sending queue. Accordingly, the *cancel rebroadcast* function removes the message corresponding to the echo from the queue.

3.4. The Protocol

The SRD protocol decision tree diagram is shown in Fig. 3. In the tail state, a vehicle *stores* all broadcasts received and rebroadcasts them with the flag *From Tail* set to true. The tail is responsible for *carrying* these messages until the connectivity in the message direction is established. The tail then *forwards* its stored messages, in this way concluding the *store-carry-forward* mechanism. Vehicles in the non-tail state have two responsibilities: (i) store all messages sent by the tail (with the *FromTail* flag set to true). This is especially important for improving the protocol robustness as we show later on; (ii) rebroadcast received messages using the Optimized Slotted 1-Persistence technique to reduce redundant retransmissions.

Transitions between the states occur as follows. A vehicle goes from the non-tail to the tail state when it goes for longer than the message direction connectivity (MDC) timer value without receiving a message retransmission from a vehicle farther in the message direction. The MDC timer duration time is defined in such a way that it considers the maximum possible time for a rebroadcast to be performed by other vehicles farther in the message direction, i.e., it must take into account message collisions, the *exponential backoff* mechanism, and the broadcast suppression technique used. The transition from tail to non-tail is triggered by the reception of a message from a vehicle farther in the message direction, as this is an indication that the cluster tail established connection with another cluster. Normally the tail has some messages stored that it needs to forward to this new cluster, so it rebroadcasts them. The vehicles from the new cluster will follow the protocol and rebroadcast these messages. Upon the receipt of an echo (i.e., a rebroadcast from a vehicle farther in the message direction) from at least one of the messages, the tail makes the transition. While the tail does not receive an echo it assumes that no vehicle in the new cluster received the messages and, therefore, the transition is not made. Once a new message from a sender farther in the message direction is received, the tail will retry this procedure. This is done to increase the protocol robustness. Note that if the tail does not have stored messages, it simply makes the transition to non-tail state as soon as a message from a vehicle farther in the message direction is received.

In order to increase robustness, every time the tail receives a message it not only stores the message but also retransmits the message with the *FromTail* set to true. By doing so, all vehicles in the range of the tail will also have a copy of that message. If the tail fails or turns off the road, eventually another vehicle will become the new tail. Since such vehicle would already have a copy of all messages received from the old tail, it can rebroadcast them whenever the MDC is reestablished. Message delivery thus is not dependent on a single vehicle. In the example shown in Fig. 4, the tail C_2 turns off the road, causing C_0 or C_1 to make the transition to the tail state. As they both have copies of C_2's messages, whichever one makes the transition will be able to retransmit the messages once the MDC is re-established. Robustness is then referred to as the

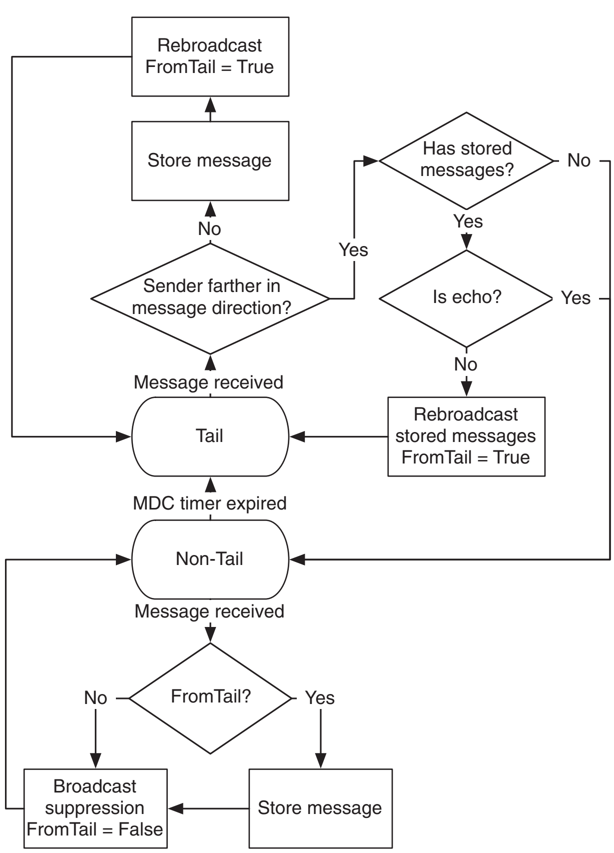


Fig. 3. SRD protocol decision tree.

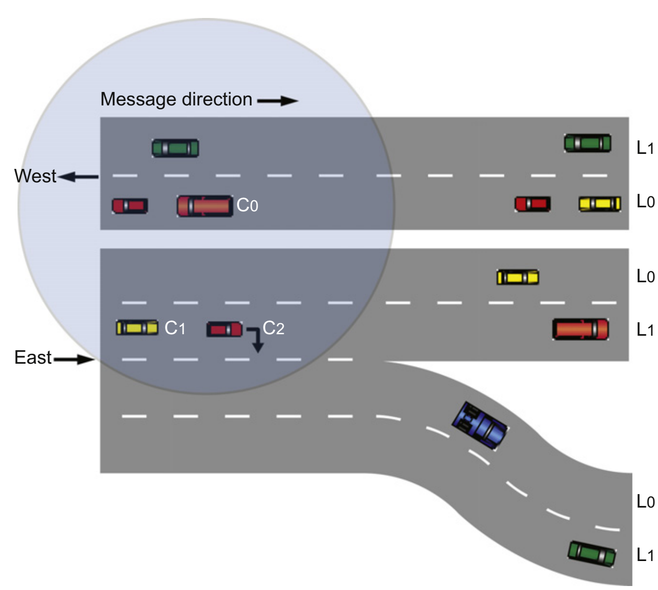


Fig. 4. Robustness motivation.

ability of the protocol to cope with highly dynamic mobility environments.

One important remark regarding broadcasting efficiency is that since the rebroadcast from the tail is always required, the tail has a higher priority in the broadcast suppression technique, in order to avoid redundant retransmissions from non-tail vehicles within that region. This priority is implemented by assigning cluster tails to the first timeslot in the Optimized Slotted 1-Persistence technique and with a smaller additional delay when compared with other non-tail vehicles also assigned to first time slot.

3.5. Defining cluster front and tail vehicles

Existing protocols such as DV-CAST require the complete knowledge of the local (1-hop neighborhood) network topology. This knowledge is usually acquired by means of *hello* messages sent by *every* vehicle periodically. These messages coexist with *event-driven* messages which are triggered upon an event such as the detection of a hazard, e.g., hard braking of cars in front. Notably, hello messages introduce undesirable side-effects such as increase of the network load, contention and collisions when not properly coordinated. All these effects together may harm the correct delivery of event-driven messages which are of primary importance in vehicular environments.

Unlike these protocols, SRD requires only the knowledge of whether the vehicle is currently the front, tail, or simply a relay vehicle in the cluster. Although the conventional hello message mechanism described above suffices to gather such information, we propose the use of a *suppressed* periodic hello message mechanism. In essence, such mechanism relies on the optimized suppression technique described in Section 3.3 to broadcast hello messages by executing the following rules:

- On a highway, assuming a pre-defined fixed message direction d, hello messages are generated by the cluster front and broadcast with periodicity λ to vehicles farther in the message direction. For the sake of explanation, let us assume d to be the easterly direction.
- Upon the receipt of a new hello message, the SRD protocol is run (as described in Fig. 3) and the message is rebroadcast with the optimized suppression technique accordingly. In order to further decrease the number of hello messages transmitted, the following modifications are introduced into the SRD protocol when dealing with hello messages:
  - Hello messages are never stored as they are simply meant to gather topology information.
  - If an event-driven message is scheduled to be rebroadcast in the suppression mechanism, any previously scheduled hello message is canceled.
- A vehicle is said to be the cluster *front* regarding the easterly direction if it does not receive a hello or event-driven message for a period longer than λ originated by a vehicle farther in the opposite (westerly) direction.
- A vehicle is said to be the cluster *tail* regarding the easterly direction if it does not receive a hello or event-driven message for a period longer than μ (set by the MDC timer) originated by a vehicle farther in the same (easterly) direction.
- The cluster front and tail vehicles of the opposite (westerly) direction are simply the cluster tail and front vehicle defined for the easterly direction, respectively.

This approach brings several advantages over typical hello message mechanisms: (i) the number of messages introduced in the network is reduced; (ii) all messages within the cluster are rebroadcast in a synchronized manner by following the optimized suppression mechanism, thereby reducing the chance of collisions; and finally, (iii) suppressed hello messages coexist more efficiently with event-driven messages, since the former are canceled upon receiving event-driven messages.

We argue that this is one possible mechanism to gather the required topology information; any other method to identify the front and tail vehicles in a cluster can be used. For instance, in [24] a protocol designed to collaboratively build an overview of the upcoming traffic in highways is presented. The front vehicle starts building a traffic map and vehicles behind it aggregate data whenever requested up to the cluster tail. Such type of mechanisms can replace our suppression hello message mechanism and still provide the required information.

3.6. Message structure

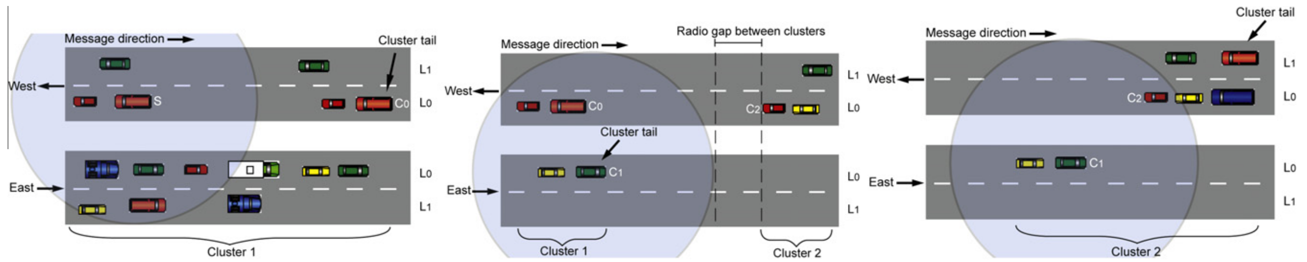
Both event-driven and hello messages have vehicle and message IDs to enable vehicles to distinguish different broadcasts and to identify rebroadcasts. An example of vehicle ID is the MAC address, while for the message ID can either be a sequence number or a timestamp of the message generation time. Such timestamp is in either case necessary in order to set an expiration time for each message and prevent old messages from being disseminated. Moreover, depending on the application-dependent size limit n for the list of stored messages, the timestamp can be used to remove the oldest messages when the limit is reached. In addition to time, the expiration mechanism can also be based on distance to prevent receiving irrelevant messages originated hundreds of kilometers away. A message could be considered expired when, for example, it reaches the end of a highway or simply after it reaches vehicles more than 10 km away from the event. For the optimized suppression technique, both message direction and geographical position of the sender are required. The former indicates to which direction the message must be propagated while the latter allows vehicles to calculate their distance with respect to the sender and choose a time slot accordingly. Finally, the *FromTail* flag utilized by SRD is included to allow vehicles surrounding the tail to store event-driven messages.

The message header structure is therefore defined by the following values: [*vehicle ID, message ID, timestamp, distance propagated, message direction, sender coordinates, from Tail flag*].

3.7. Basic operation example

The basic operation of the SRD protocol is shown in Fig. 5. In Fig. 5(a), a message is sent towards the easterly direction. When a message coming from the west and disseminated by SRD reaches and is rebroadcast by vehicle S, all vehicles except for the tail simply rebroadcast the message generated using the proposed broadcast suppression technique. When a non-tail vehicle receives a message from another non-tail vehicle that is farther in the message direction, it simply drops the message and cancels (suppresses) any previously scheduled transmission in case the message received is an echo. All vehicles help disseminating the message, regardless of the direction they are moving. Whenever the broadcast message reaches the tail (C_0 in Fig. 5(a)), the cluster tail stores the message and rebroadcasts it with the *FromTail* flag set to true. In this way, all non-tail vehicles that hear the rebroadcast from the cluster tail also store the message.

A change in the cluster tail is shown in Fig. 5(b), in which C_1 listens to a rebroadcast from the tail C_0. Even though C_1 realizes that the sender was not farther in the message direction, the message is stored as it comes with the *FromTail* flag set to *True*. Following the protocol, C_1 rebroadcasts it using the broadcast suppression technique. This rebroadcast is needed since C_1 does not yet know whether it is the new tail. C_0 then receives this retransmission and verifies that the sender is farther in the message direction.



(a) time = t_0 (b) time = $t_0 + x_1$ (c) time = $t_0 + x_1 + x_2$

Fig. 5. Protocol description.

Consequently, it retransmits all stored messages and starts the transition procedure to the non-tail state. This retransmission is done to cover two possibilities. First, there could be a gap after C_1 farther in the message direction and C_1 would become the new cluster tail (as shown in Fig. 5(b)). In this case, the rebroadcast is done to guarantee that the new tail has a copy of all messages from the old tail (C_0). In the second case (not shown in the figure), the gap does not exist, i.e. there is a vehicle in the range of the C_1 that is not in the range of C_0. The retransmission in this case will cause C_1 to relay all messages to this farther vehicle and consequently to all others that it might be connected to. In either case, upon the receipt of C_1 rebroadcasts, C_0 concludes the transition to non-tail state.

As C_1 is moving farther in the message direction, at some point it enters in the communication range of C_2, reaching a new cluster, as shown in Fig. 5(c). When this happens, C_1 eventually receives a message from C_2. As C_2 is farther in message direction, C_1 starts the transition from tail to non-tail state rebroadcasting every stored message it carries. At this point, C_2 and all non-tail vehicles within its cluster will rebroadcast the messages received in order to spread them to other vehicles farther in the message direction. When the C_1 receives one of these rebroadcasts, it concludes the transition to non-tail state.

4. Performance evaluation

In this section, we present the performance evaluation of the SRD protocol carried out by means of simulations with OMNeT++ 4.1.[2] We consider four protocol versions: SRD and DV-CAST; and their respective suppression techniques used for dense scenarios, namely, Optimized Slotted 1-Persistence and Slotted 1-Persistence. Our goal is to evaluate SRD under various vehicle scenarios and compare it directly with DV-CAST, which is the existing protocol that also focus on directional broadcasting in both dense and sparse highways. The separate evaluation of the suppression techniques serves to assess the actual gain of the store-carry-forward models employed by both SRD and DV-CAST in sparse networks.

In our simulations, we utilize the MiXiM Framework[3] and adjust the available implementation of the IEEE 802.11b protocol to comply with basic specifications of the 802.11p version. In the MAC layer, we set the bit rate to 6 Mbit/s, the Contention Window (CW) to values between 15 and 1023, the slot time to 13 μs, the SIFS to 32 μs, and the DIFS to 58 μs. In the physical layer, we operate on the 5.9 GHz frequency band, with 10 MHz of bandwidth. Based on estimates, we set the transmission power to 50 mW to achieve approximately 350 m of interference range and 176 of transmission range, assuming the Friis free space path loss (FSPL) propagation model with exponent α equal to 3.5.

[2] http://www.omnetpp.org/.
[3] http://mixim.sourceforge.net.

Table 1
Simulation parameters

Physical layer	Frequency band	5.9 GHz
	Bandwidth	10 MHz
	Transmission power	50 mW
	FSPL exponent α	3.5
Link Layer	Bit Rate	6 Mbit/s
	CW	[15,1023]
	Slot Time	13 μs
	SIFS	32 μs
	DIFS	58 μs
Supression Mechanism	*st*	5 ms
	NS_{std}	3
	NS_{opt}	6
	D_{max}	1 ms
	MDC	0.1 s
	Hello size	24 Bytes
	Hello Frequency	1 Hz
Scenarios	message size	2312 Bytes
	Message frequency	0.5 Hz
	# Runs	50

For the suppression technique mechanisms, we set *st* to 5 ms. We define the total number of time slots for the Slotted 1-Persistence protocol (NS_{std}) to 3 and for the Optimized Slotted 1-Persistence protocol (NS_{opt}) to 6 (3 slots for each road direction), except in Section 4.4 where the protocols are evaluated for different values of *NS*. The value set for the Slotted 1-Persistence protocol is based on [10] while the value set for the Optimized Slotted 1-Persistence protocol is based on preliminary simulation studies. For the maximum additional delay D_{max}, we use 1 ms. The MDC timer defined in the SRD protocol is set to expire after 0.1 s. For the SRD and DV-CAST protocols we set the size of hello messages to 24 bytes and they are generated with 1 Hz frequency. Also for both protocols, for the sake of simplification we do not set size limits for the lists which keep track of the most recent messages IDs and which store the messages in the store-carry-forward mechanisms.

For all simulation scenarios the message size is the same, 2312 bytes, the maximum allowed by the 802.11p standard. New messages are generated every 2 s, i.e., the message frequency is 0.5 Hz. Each message is generated by the farthest vehicle in one end of the road. For each simulation scenario 50 runs are executed. Table 1 contains a summary of the simulation parameters common to all simulation scenarios.

Our evaluation considers the following metrics:

- ***Delivery ratio***: the percentage of messages generated by the farthest vehicle in one end of the road which fully propagate and are received by a vehicle in the extreme opposite end of the road. Ideally, dissemination protocols must achieve a delivery ratio percentage close to 100%.

- ***Network load***: to evaluate this metric, two sub-metrics are defined: the total number of transmissions and the total number of receptions performed on average by an arbitrary vehicle. These values are normalized by the total simulation time. In order to be efficient, protocols must have a low number of transmissions and receptions during their dissemination.
- ***Delay***: the total time taken for a message to propagate from one end to the other of the road length. This is particularly important for critical information that must be disseminated as rapidly as possible.

4.1. Controlled scenarios

We start our simulation campaign by studying the performance of the protocols for various controlled scenarios. By *controlled* we mean that we build the vehicle distribution along the road in such a way that it allows us to understand with precision what to expect from the protocols. This is important since we want to test the protocols in specific road conditions which are hard to reproduce in most traffic simulators. In particular, we guarantee for these scenarios that the maximum theoretical delivery ratio is 100%. This does not occur for traffic simulator scenarios as we report in Section 4.2. In Section 4.1.1, static scenarios with equally spaced vehicles are used to evaluate the scalability of the protocols in highly congested roads with increasing densities. In this way, we simulate well connected networks in intense traffic jams where vehicles would practically not move. Following, in Section 4.1.2 we concentrate on evaluating the protocols in basic mobility scenarios where protocols must deal with both well connected and sparse networks. Here, we focus in particular on cases where store-carry-forward mechanisms should overcome gaps between vehicle clusters.

4.1.1. Static scenarios

In this first set of scenarios, we study the performance of the protocols for various traffic densities. To allow that, we simulate a two kilometer road with static vehicles placed in both easterly and westerly directions, with each direction comprising two lanes. We consider congested scenarios where vehicles are equally spaced in such a way that there is no radio gap between them. We vary the number of vehicles from 20 to 100 vehicles/km/lane. Each simulation run has a duration time of 60 s.

In terms of network load, with an increase in density the number of receptions and transmissions performed on average is generally increased, as shown in Figs. 6(a) and (b). This is expected as more vehicles are assigned to individual time slots and thus rebroadcasting. The results also show a clear higher number of receptions and transmissions, namely, the double of receptions and transmissions performed by DV-CAST when comparing with SRD. This is in great part explained by the higher number of periodic hello messages transmitted. On the other hand, when comparing their respective strategies employed to cope with dense scenarios, the Optimized Slotted 1-Persistence presents a slightly higher load when compared with Slotted 1-Persistence. Although Optimized Slotted 1-Persistence relies on the double number of time slots to decrease the number of vehicles rebroadcasting, the policy adopted to cancel rebroadcasts is more strict when compared with Slotted 1-Persistence as it only allows a transmission to be suppressed when an echo is received by other vehicles farther in the message direction. For instance, depending on the vehicle distribution, some vehicles positioned in the low priority direction will not cancel their rebroadcasts if they hear earlier from other vehicles in the high priority direction which are not located farther in the message direction.

Although our policy to suppress transmissions leads to a higher load with Optimized Slotted 1-Persistence in comparison with Slotted 1-Persistence, the delivery ratio is clearly improved, as shown in Fig. 6(c). More specifically, the delivery ratio achieved with Optimized Slotted 1-Persistence is near 100% for all densities. Such performance is also valid for SRD. The other protocols, namely, DV-CAST and Slotted 1-Persistence present a low delivery ratio in low densities. Because of the lack of mechanisms to cope with simultaneous broadcasts, collisions become severe for these protocols specially in low densities. However, as the density increases their performance generally improves. This can be explained by the higher probability that at least one rebroadcast is successful when more vehicles are present in each time slot. When higher densities are considered, the delivery ratio of DV-CAST varies from 80% to 95% whereas Slotted 1-Persistence delivers nearly 100% of the messages. This can be reasoned by the higher and asynchronous number of messages transmitted by DV-CAST due to the use of hello messages, which leads to a higher number of collisions.

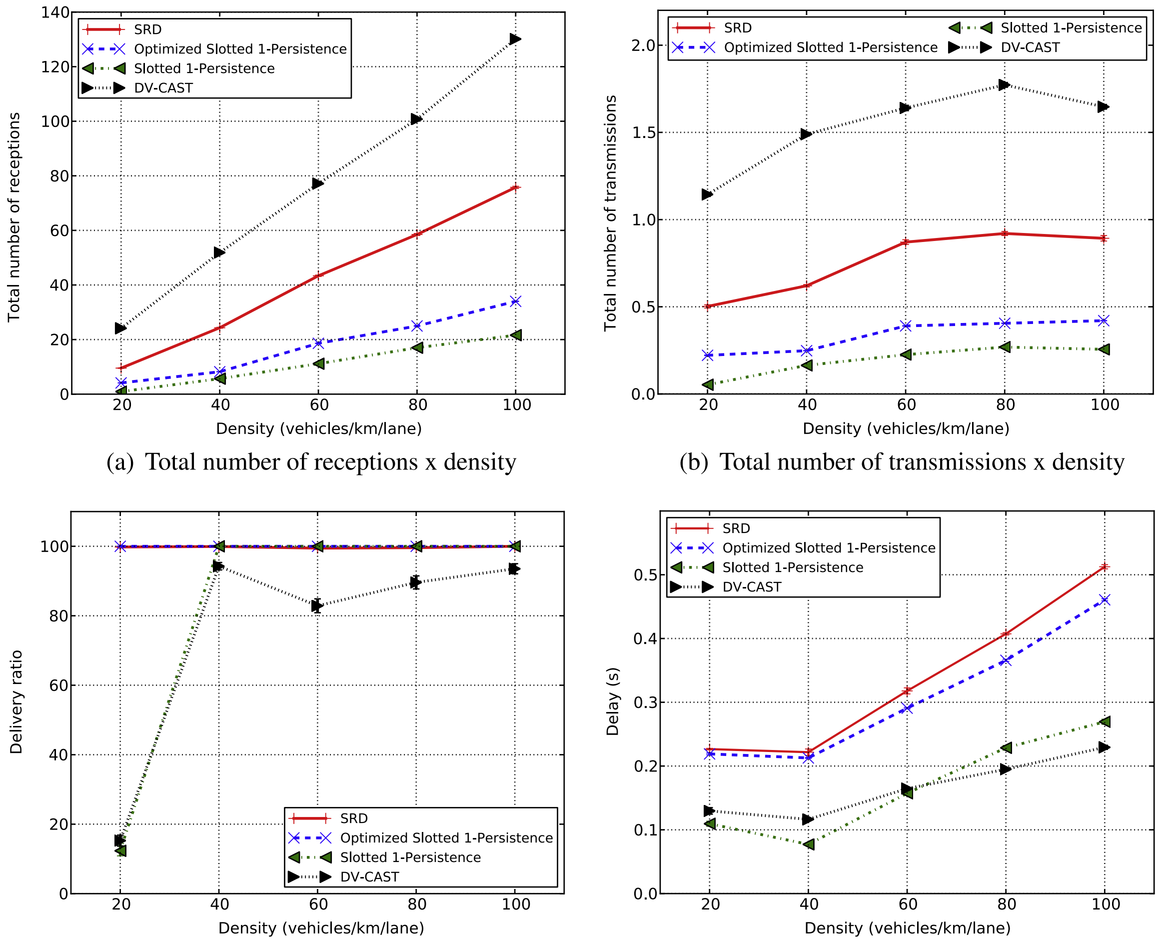
Fig. 6(d) shows the performance with respect to the end-to-end delay. The delay is higher when the density increases for all protocols. This is due to the higher number of vehicles transmitting near simultaneously in an individual time slot, which leads to a higher number of collisions and contention period in the MAC protocol layer. As expected, the delay is higher with SRD and Optimized Slotted 1-Persistence, since some rebroadcasts are performed by vehicles in the low priority direction in later time slots. Nevertheless, an end-to-end delay of 0.5 s for the complete propagation of a message in a road length of 2 kilometers might be low enough to suit most vehicular applications such as for the awareness of an accident ahead on the road.

Overall, SRD and Optimized Slotted 1-Persistence outperform DV-CAST and Slotted 1-Persistence with regard to delivery ratio, which is the ultimate goal of dissemination protocols. In order to achieve such performance, we decrease the number of vehicles transmitting in an individual time slot by relying on a higher number of time slots. The trade-off is the higher end-to-end delay in comparison with the other protocols. Finally, the suppressed hello periodic hello message mechanism employed by SRD help decrease by half the number of receptions and transmissions performed by DV-CAST.

4.1.2. Mobility scenarios

In this section, we consider three basic mobility scenarios which dissemination protocols must address, namely, a well-connected dense network, a network with a radio gap in one road direction, and a network with radio gaps in both road directions. We simulate a two kilometer highway with two lanes per road direction and four vehicle clusters (groups). Lanes are 4 m wide with a 10 m space between the directions. In Scenario 1, all four lanes are very busy, with 100 vehicles/km/lane. In each lane there is a group of 250 vehicles separated by 10 m. The initial state is shown in Fig. 7(a). Vehicles move at speeds between 2 and 2.5 km/h and there is always connectivity between the groups during the simulation time. Scenarios 2 and 3 simulate situations with radio gaps between vehicles clusters. In scenario 2, shown in Fig. 7(b), there is a 500 m gap between groups 1 and 2 in such a way that they cannot communicate directly. In this scenario, each group has a density of 20 vehicles/km/lane and vehicles move at speeds between 115 and 120 km/h. In scenario 3 (Fig. 7(c)) the gap also exists but there are no vehicles moving in the opposite direction in the initial state. To bridge the gap, vehicles moving in the opposite direction use the store-carry-forward mechanism. Vehicle densities and speeds are the same as in scenario 2. Although there is a small vehicle speed variation in these scenarios, no overtaking or lane changing are simulated. Moreover, in each simulation run the duration time is set to 60 s and vehicles move at intervals of 0.1 s.

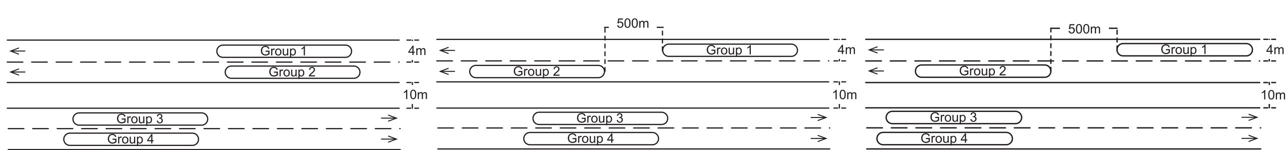
Figs. 8(a) and (b) show the number of receptions and transmissions for each scenario, respectively. For every scenario, the



(a) Total number of receptions x density (b) Total number of transmissions x density

(c) Delivery ratio x density (d) Delay x density

Fig. 6. Performance evaluation for controlled static scenarios with 95% confidence intervals.



(a) Scenario 1 (b) Scenario 2 (c) Scenario 3

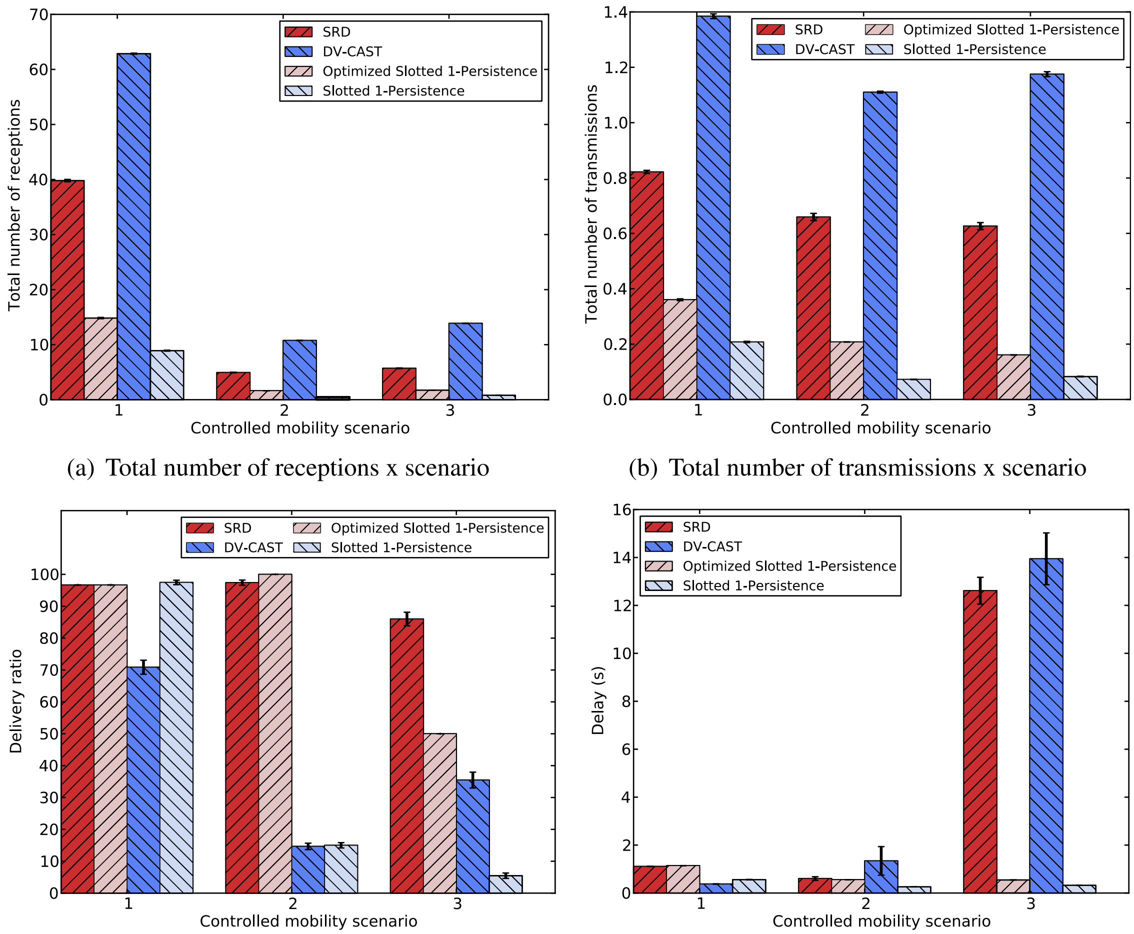
Fig. 7. Mobility scenarios.

number of receptions and transmissions performed by DV-CAST is notably higher when compared with other protocols. Furthermore, Optimized Slotted 1-Persistence presents a slight increase when compared with Slotted 1-Persistence. The network load is lower for scenarios 2 and 3 due to their lower densities. These results are in line with the results obtained for the static scenarios and, therefore, their rationales are analogous.

With regard to delivery ratio in Fig. 8(c), all protocols with the exception of DV-CAST achieve nearly 100% in scenario 1 where a well-connected network is present. In scenario 2, SRD and Optimized Slotted 1-Persistence present nearly 100% of delivery ratio whereas the percentages with DV-CAST and Slotted 1-Persistence are limited to 15%, which can also be explained by the effect of collisions in low densities. Finally, in scenario 3 the store-carry-forward model of each protocol is evaluated. While SRD achieves nearly 90% of delivery ratio against 50% with Optimized Slotted 1-Persistence, the percentage with DV-CAST is limited to nearly 40% against 5% achieved by Slotted 1-Persistence.

The performance results for end-to-end delay are shown in Fig. 8(d). As expected, the delay values for scenario 1 and 2 are considerably lower than for scenario 3. In scenario 3, vehicles have to store, carry, and forward all messages from group 2 to group 1 in order to overcome the radio gap. Therefore, the delay is dependent on the speed of vehicles, which in this case leads to over 10 extra seconds compared to the remaining scenarios. In scenario 1, because of the use of later time slots for the low priority direction, SRD and Optimized Slotted 1-Persistence present higher delays than DV-CAST and Slotted 1-Persistence. This is also valid for scenario 2 with respect to Slotted 1-Persistence. However, DV-CAST presents an even higher delay than other protocols most likely because of the higher number of collisions caused by hello messages.

While the results with respect to the network load correspond to the results presented for static scenarios, the delivery ratio gain when employing the store-carry-forward models in scenarios with separate vehicle cluster is evident. More specifically, the gain with SRD is of 40% and with a maximum delivery ratio of nearly 90%,



(a) Total number of receptions x scenario

(b) Total number of transmissions x scenario

(c) Delivery ratio x scenario

(d) Delay x scenario

Fig. 8. Performance evaluation for controlled mobility scenarios with 95% confidence intervals.

whereas DV-CAST improves the delivery ratio in 35% and to a maximum of only 40%.

4.2. Traffic Simulator Scenarios

After analyzing the performance of the protocols in controlled vehicle distributions, we now consider realistic vehicle distributions generated by professional traffic microsimulators, namely Quadstone Paramics 5.2 [25] and SUMO 0.11.1 [26]. The reason for relying on two different traffic microsimulators is that SUMO has been preferred in a later stage of this work because of its facilities to export vehicle traces to other software such as network simulators. Both simulators are widely used by researchers and professionals and support features such as overtaking, lane changing, and rely on well-known car-following mobility models such as Krauß [27] and Intelligent Driver Model (IDM) [28]. In particular, the Krauß model is used in our simulations.

Three scenarios are considered in this set of simulations: a well-connected network (scenario 1), a sparse network (scenario 2), and a network combining both sparse and dense networks (scenario 3). Scenarios 1 and 2 are created with SUMO. We build a 10 km straight highway with two lanes per road direction. Lanes are 4 m wide with a 10 m space between the directions. With this road, we differentiate the two scenarios by the traffic demands assigned to each one, namely, a moderate traffic flow generated for scenario 1 which leads to a density of 7.5 vehicles/km/lane and for scenario 2 a low traffic flow which leads to a density of 2.5 vehicles/km/lane. The speed at which vehicles move varies from 80 km/h to 120 km/h. The vehicle generation rate remains constant for both scenarios in such a way that the average density assigned remains also constant. Moreover, in each simulation run the duration time is set to 300 s and vehicles move at intervals of 1 s (standard frequency set in SUMO).

Scenario 3 is shown in Fig. 9. We simulate a 10 km road with two lanes in each direction. In one direction, vehicles are sparsely distributed with density varying from 2 to 10 vehicles/km/lane while in the other direction a traffic jam is induced near a junction with a rapid increase in density from 20 to 40 vehicles/km/lane. The junction point is located at the center of the road (5 km) and it divides the road into two other roads with one lane each and with moderate traffic (10 to 20 vehicles/km/lane). The distribution of vehicles is generated by the Quadstone Paramics 5.2 traffic simulator executed with the CeeJazz plug-in. In the sparse and moderate traffic lanes, vehicles move on average at 120 km/h whereas in the section with an induced traffic jam, their speed drops rapidly from 120 km/h to 5 km/h. In this scenario, in each simulation run the duration time is set to 300 s and vehicles move at intervals of 0.5 s (standard frequency set in the CeeJazz plug-in).

All the three scenarios described have the particularity that vehicles are generated (allocated) in the simulation when they enter the road in one extreme and are deallocated when they reach the other extreme end of the road direction. Because we need to keep track of which messages are generated and successfully propagated along the complete road, we place one static network node

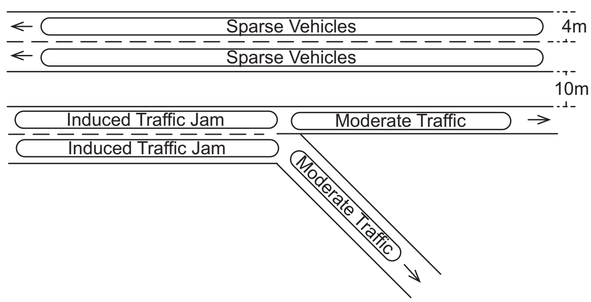


Fig. 9. Traffic simulator scenario 3.

in each end of the road: one node to generate (broadcast) messages, e.g., a crashed vehicle, positioned in the foremost position in the westerly direction; and another node, e.g., a road-side unit, to collect all messages and generate statistic results of the simulation. For this reason, the maximum theoretical delivery ratio is not guaranteed to be 100%, since at the moment when a message is generated there might be no vehicles within the transmission range to receive and further propagate the message along the road. In addition, in these traces vehicles perform lane changing, overtaking, and therefore change their order during the simulation. This realistic behavior helps us understand the level of robustness in a wide variety of traffic situations.

Figs. 10(a) and (b) show the performance of the protocols with regard to the network load. For all scenarios, the results present a similar pattern found in the controlled mobility scenarios with DV-CAST performing the highest amount of receptions. When higher densities are considered in scenario 1 and further in scenario 3, higher values of reception are indicated. In scenario 2, because the network is very sparse, there is no multi-hop connectivity between any vehicle. The consequence is that the numbers for reception and transmission are equivalent for both SRD and DV-CAST protocols, whereas for the remaining protocols the values are nearly zero due to the lack of store-carry-forward mechanisms. In particular, the number of transmissions is higher when employing SRD in scenario 3.

The results regarding the delivery ratio are shown in Fig. 10(c). In scenario 1, where a well-connected network is present, protocols should achieve nearly 100%. This is the case for both SRD and Optimized Slotted 1-Persistence. In contrast, DV-CAST and Slotted 1-Persistence perform very poorly in the best case reaching 5% of delivery ratio. Because of the characteristics of scenario 2 and how we set up the simulation, the maximum delivery ratio is limited to a much lower value. In fact, because there is no multi-hop connectivity between vehicles and because hello messages which are constantly sent by each vehicle can collide with the messages generated by the static network node, the probability that a message is correctly received by any vehicle to be further disseminated along the road is much lower than in the other scenarios. For this scenario both SRD and DV-CAST achieve near 10% of delivery ratio. The remaining protocols present delivery ratio of zero, as expected in a very sparse scenario.

The results for scenario 3 show a high difference in performance when comparing SRD and Optimized Slotted 1-Persistence with DV-CAST and Slotted 1-Persistence. SRD achieves 80% of delivery ratio against 65% achieved by Optimized Slotted 1-Persistence – 15% of gain with the use of the store-carry-forward model employed by SRD. In contrast, DV-CAST and Slotted 1-Persistence present delivery ratio of only approximately 4% and 2%, respectively. Such difference explains the higher number of transmissions performed by SRD in comparison with DV-CAST, since with a higher delivery ratio more vehicles rebroadcast along the road.

Fig. 10(d) shows the results with respect to the end-to-end delay. For scenario 1, DV-CAST presents a much higher delay (over 150 s) when compared with the remaining protocols. This shows that Slotted 1-Persistence employed by DV-CAST does not disseminate the messages properly, since one would expect a quick message dissemination in a well-connected network. The high delay values indicate that DV-CAST can only deliver a few messages by using its store-carry-forward mechanism. In scenario 2, because of the lack of multi-hop connectivity, the delay is directly related to the speed at which vehicles move. Thus, the average of both SRD and DV-CAST is near 250 s. Finally, in scenario 3 the delay for SRD is higher than for other protocols. This is explained by the higher delivery ratio achieved when using its store-carry-forward mechanism, with some messages arriving later in the simulation.

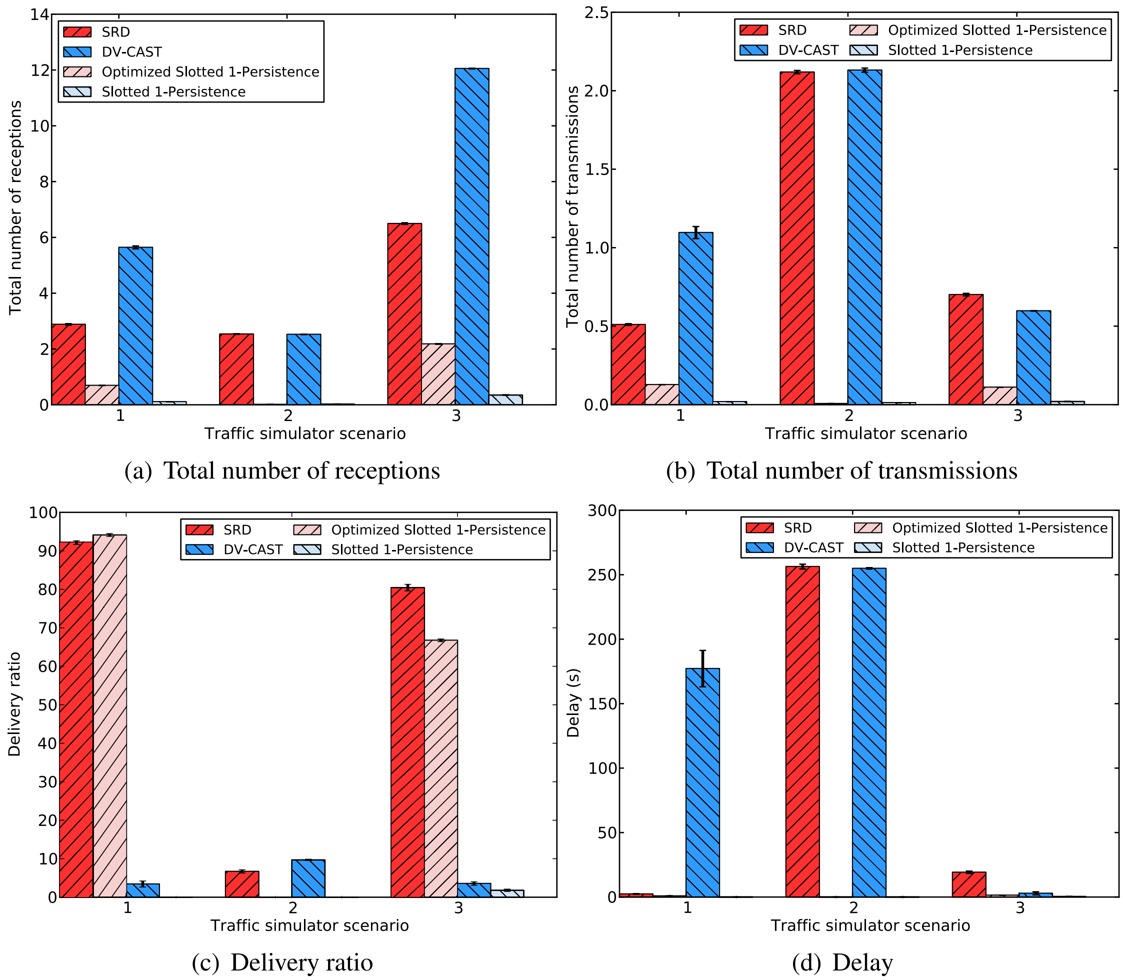
This evaluation shows the significant improvement when using SRD and Optimized Slotted 1-Persistence over DV-CAST and Slotted 1-Persistence with regard to delivery ratio. Notably, the number of receptions with DV-CAST is higher than with SRD even with such poor results with respect to delivery ratio.

4.3. Robustness Test

In this section, we assess the mechanism which SRD employs to improve robustness in highly dynamic vehicle distributions. As mentioned in Section 3.4, we define robustness as the ability of each protocol to cope with rapid changes in the network topology. Our goal is to evaluate the effects on the delivery ratio when running the protocols in scenarios including, but not limited to, vehicles leaving the road and vehicle crashes. In particular, we evaluate the ability of each protocol to cope with the situation shown in Fig. 4 where the cluster tail leaves the road and the messages stored must still be forward to other vehicles ahead on the road despite such change in the network topology. Such situation is present in traffic simulator scenario 3, where a junction is present. However, since there is a constant change in topology in all traffic simulator scenarios, the impact on robustness becomes hard to be assessed. Therefore, for this assessment we reuse the scenario 3 among our controlled mobility scenarios. The reason for choosing this scenario is that it is the only controlled scenario where a radio gap is present and thus the only scenario possible to reproduce the situation described. In addition, only SRD and DV-CAST are evaluated since their suppression mechanism alone cannot handle radio gaps.

Identically to what is done in the controlled scenarios, messages are generated by the foremost vehicle in the westerly direction. We simulate the tail vehicles of the cluster formed by groups 2, 3, and 4 (Fig. 7(c)) leaving the road by turning off their radios. After the tail is placed out from the road, this process is repeated and the new tail leaves the road after a pre-defined time. We define three different frequencies (intervals) at which vehicles leave the road: at every 5, 10, and 15 s. In this way, at the highest frequency, i.e., the current tail leaving the road at every 5 s, we simulate a highly dynamic environment.

The performance regarding the delivery ratio is shown in Fig. 11. We compare the results obtained in the simulations with vehicles leaving the road at each frequency side-by-side with the previous results obtained in the original controlled mobility scenario 3. The results for this simple scenario validate the mechanism employed by SRD by showing an unaltered performance for all frequencies when compared with the previous results in scenario 3. In contrast, DV-CAST is affected with a decrease in delivery ratio as the frequency increases. When tail vehicles leave the road at a rate of one at every 5 s, the delivery ratio decreases from 35% to 25%. This deterioration is explained by the reliance on a single vehicle by DV-CAST to store, carry, and forward messages in sparse



(a) Total number of receptions (b) Total number of transmissions

(c) Delivery ratio (d) Delay

Fig. 10. Performance evaluation for traffic simulator scenarios with 95% confidence intervals.

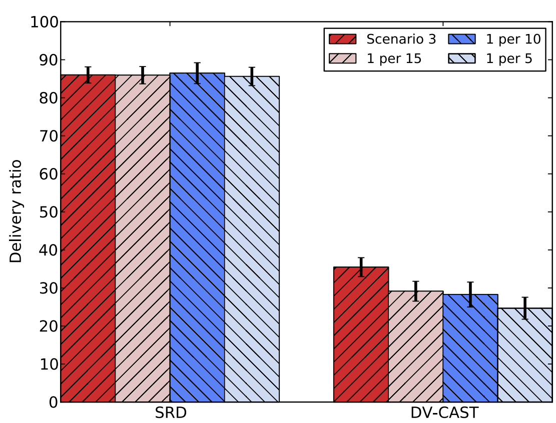


Fig. 11. Delivery ratio x frequencies at which vehicles leave the road with 95% confidence intervals.

networks. As explained in Section 3.4, with SRD all vehicles which receive a message with the flag *FromTail* equal to true will store the message and act as back-up vehicles in case the tail vehicle leaves the road or fail, thereby increasing robustness in such usual road scenarios.

With regard to the network load, the results follow the same pattern as the ones obtained for the delivery ratio. More specifically, in the results obtained with DV-CAST there is a decrease in both receptions transmissions when higher frequencies for the cluster tail leaving the road are considered. The end-to-end delay remain unaltered. With respect to SRD, all results are practically analogous with negligible variations compared to the ones obtained for the controlled mobility scenario 3. For these reasons, such results are not depicted in this section.

4.4. Effects of the total number of time slots

In this section we assess the impact of choosing various values for the parameter regarding the total number of time slots (*NS*). Notably, this parameter influences directly on the performance of the suppression mechanisms employed by SRD and DV-CAST protocols as it defines the number of vehicles assigned to each time slot and thus the number of vehicles rebroadcasting near simultaneously. For this evaluation, we run both SRD and DV-CAST with *NS* values varying from 1 to 10. However, because SRD always rely on an equal number of time slots for each road direction, only even numbers from this range are evaluated for SRD. We choose traffic simulator scenario 3 for this simulation, due to the presence of a high dynamic road environment which yields to a wide variety of situations. Since the suppression mechanisms employed by both SRD and DV-CAST have already been evaluated separately in previous sections, we omit their assessment here.

We start our discussion with Fig. 12(c) where the delivery ratio for each protocol is shown for increasing values of *NS*. While SRD improves its performance with higher values for the number of time slots, DV-CAST performs poorly and reaches delivery ratio of nearly zero percent when *NS* is equal to 4, going down to zero in the remaining values. As explained previously, DV-CAST presents poor results when there are few vehicles assigned to each time slot, which indicates that collisions have a high impact on its functioning. On the other hand, the optimizations proposed in SRD clearly deals efficiently with different values for *NS*. More specifically, with fewer vehicles assigned to each time slot fewer rebroadcasts are performed. With the optimizations proposed to tackle collisions and increase robustness by being more strict when canceling rebroadcasts, fewer collisions are also present which in turn allows for a better performance in delivery ratio. As we emphasized before, because of the characteristics of the simulation in traces generated by traffic simulators, 100% of delivery ratio is not possible in this scenario.

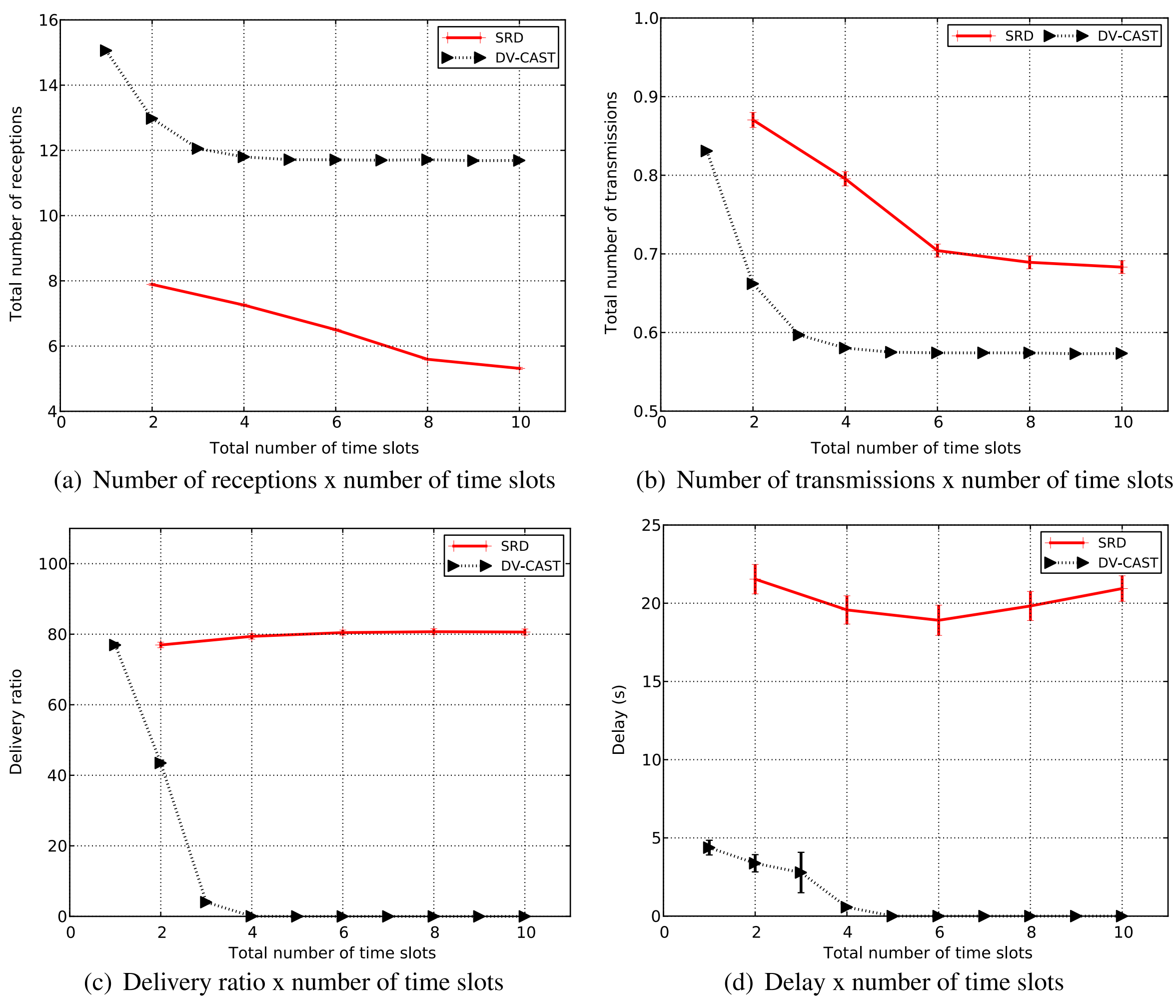
Fig. 12(a) and (b) shows the results for the number of transmissions and receptions. Generally, both values decrease with a higher number of time slots. While with SRD this result is directly explained by the fewer number of vehicles rebroadcasting in each time slot, the lower values with DV-CAST are directly related to the poor delivery ratio which yields obviously in fewer transmissions and receptions. After the delivery ratio reaches zero percent for DV-CAST, the values for the number of transmissions and receptions remain stable. This might indicate that the message propagation ends (e.g., due to collisions) at similar points in the end-to-end path, which makes the results for the network load very similar with predominantly hello messages being sent an received.

Finally, the performance regarding the end-to-end delay is shown in Fig. 12(d). With SRD, the decrease in delay up to *NS* equal to 6 is a result of the presence of fewer collisions per time slot. With fewer collisions, the probability that the farthest vehicle from the sender succeeds in rebroadcasting is higher, which decreases the multi-hop end-to-end delay. After *NS* equal to 6, the delay start to increase. This is an indication that the earliest time slot is not always utilized due to the absence of vehicles assigned to it. Because of the higher delivery ratio and the consequent higher use of the store-carry-forward mechanism, the delay values for SRD are generally higher than for DV-CAST. The delay also decreases with DV-CAST also due to the fewer number of messages that travels the complete road.

From the results above, assigning *NS* to 6 seems to be the optimal value for this scenario. We can conclude that based on the delay and the results achieved for the network load and delivery ratio.

4.5. Effects of hello messages

Our final evaluation concerns the impact that hello messages have on the performance of SRD. As described in Section 3.5, SRD relies on what we call a *suppressed* periodic hello message mechanism. However, a beaconing [29] mechanism is expected to coexist with other systems in a vehicle. Such mechanism sends out



(a) Number of receptions x number of time slots

(b) Number of transmissions x number of time slots

(c) Delivery ratio x number of time slots

(d) Delay x number of time slots

Fig. 12. Performance evaluation in traffic simulator scenario 3 for different total number of time slots with 95% confidence intervals.

periodic messages called *beacons* which have the same function as hello messages and contain information such as geographical location, speed, and acceleration. Therefore, it is also important to evaluate how SRD behaves when employed with a regular *uncompressed* hello message mechanism. To accomplish this evaluation, we remove the mechanism used by SRD to gather the minimum topology information required, namely, the MDC timer and the use of *suppressed* hello messages. Instead, we insert an equivalent hello message mechanism as employed by DV-CAST. Thus, hello messages contain only the vehicle ID, message ID, timeStamp, and the sender coordinates in order to derive which vehicles are the cluster front and tail. We compare this SRD version with SRD using suppressed hello messages and also with DV-CAST. We choose traffic simulator scenario 3 for this evaluation.

Fig. 13(a) and (b) shows the results for the network load. With an un-suppressed scheme, the higher numbers of transmissions and receptions when compared with SRD running the suppressed mechanism are evident. Such numbers are also higher compared to DV-CAST, which is explained by the higher delivery ratio as shown in Fig. 13(c). In fact, the use of un-suppressed hello messages reduces the delivery ratio in approximately 5%. However, compared with DV-CAST such loss is negligible. Finally, Fig.13(d) shows that with an un-suppressed hello message scheme, lower delay values are present. Since the delivery ratio achieved by the protocols diverge, the assessment of the end-to-end delay becomes difficult. For instance, if more messages succeed in propagating the complete road path, some messages could arrive later due to radio gaps and thus increase the end-to-end delay average achieved by a certain protocol. This can be one reason for such lower values. Another reason can be that with un-suppressed hello messages, vehicles are able in this scenario to estimate their current role in the cluster (front or tail) more quickly and accurately leading in this way to a quicker propagation along the road. Further study is necessary to validate this assumption.

Overall, SRD with an un-suppressed hello messages presents a similar performance in terms of delivery ratio when compared with SRD running its suppressed hello message mechanism. The main difference lies in the higher network load when using the un-suppressed mechanism.

5. Conclusion and future work

In this paper we have presented a dissemination protocol suitable for both sparse and dense vehicular networks. The use of suppression techniques has been motivated and employed in dense networks while the store-carry-forward communication model has been used in sparse networks. The designed protocol is both *simple* and *robust*. We have proposed an optimized suppression technique which is based on the Slotted 1-Persistence [4]. Furthermore, because SRD requires only limited local topology information, we have presented an efficient periodic hello message mechanism in which only a subset of vehicles is required to participate. Our simulation results show that SRD outperforms DV-CAST in terms of delivery ratio for the diverse set of scenarios considered and introduces a lower load into the network. In addition, SRD presents better performance with regard to robustness in highly dynamic scenarios where vehicles move to different roads frequently.

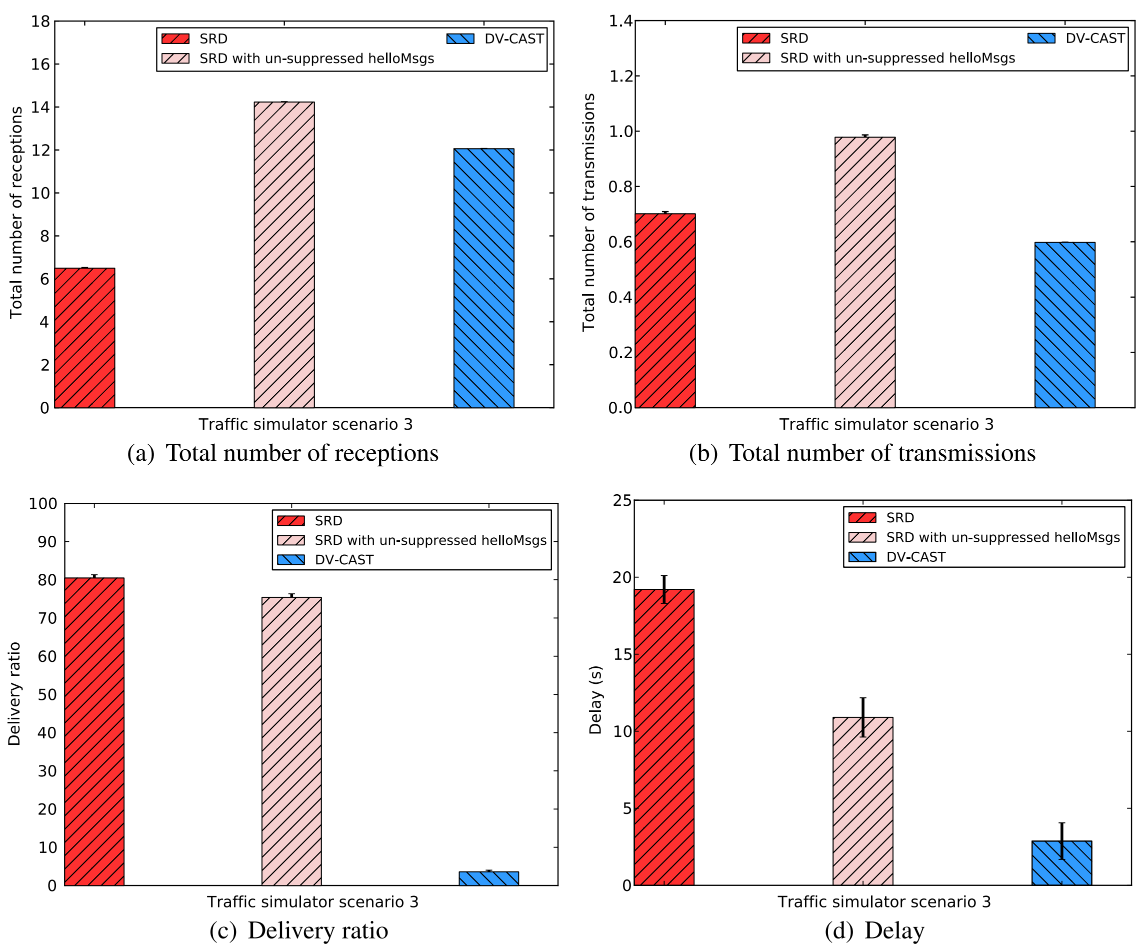


Fig. 13. Performance evaluation in traffic simulator scenario 3 for SRD running with uncompressed hello messages with 95% confidence intervals.

In future work, we will aim to improve the performance with regard to end-to-end delay and use power control mechanisms to further decrease the network load in dense scenarios. In addition, we plan to extend our protocol to make it suitable to urban environments.

Acknowledgments

The authors express their sincere gratitude to Martijn van Eenennaam, Georgios Karagiannis and Wouter Klein Wolterink for their collaboration and for the discussion of developing optimizations for broadcast suppression techniques.

References

[1] H. Hartenstein, K.P. Laberteaux, Topics in ad hoc and sensor networks - a tutorial survey on vehicular ad hoc networks, IEEE Communications Magazine 46 (6) (2008) 164–171. doi:10.1109/MCOM.2008.4539481.
[2] U. Lee, M. Gerla, A survey of urban vehicular sensing platforms, Computer Networks 54 (4) (2010) 527–544, doi:10.1016/j.comnet.2009.07.011.
[3] S.Y. Ni, Y.C. Tseng, Y.S. Chen, J.P. Sheu, The broadcast storm problem in a mobile ad hoc network, in: Proceedings of the 5th annual ACM/IEEE International Conference on Mobile Computing and Networking, ACM, New York, NY, USA, 1999. 152–162.
[4] N. Wisitpongphan, O.K. Tonguz, J.S. Parikh, P. Mudalige, F. Bai, V. Sadekar, Broadcast storm mitigation techniques in vehicular ad hoc networks, IEEE Wireless Communications 14 (6) (2007) 84–94, doi:10.1109/MWC.2007.4407231.
[5] S.Jain, K.Fall, R.Patra, Routing in a delay tolerant network, in: Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, ACM, New York, NY, USA, 2004, pp. 145–158. Available from: <http://portal.acm.org/citation.cfm?id=1015484>.
[6] C.M. Huang, K.c. Lan, C.Z. Tsai, A survey of opportunistic networks, in: 22nd International Conference on Advanced Information Networking and Applications – Workshops (aina workshops 2008), IEEE, 2008, pp. 1672–1677. doi:10.1109/WAINA.2008.292.
[7] A.Agarwal, D.Starobinski, T.D.C. Little, Exploiting downstream mobility to achieve fast upstream message propagation in vehicular ad hoc networks, Mobile Networking for Vehicular Environments. Anchorage, AK, 2007.
[8] N. Wisitpongphan, F. Bai, P. Mudalige, O.K. Tonguz, On the routing problem in disconnected vehicular ad hoc networks, in: Proceedings IEEE Infocom 2007.
[9] R.S. Schwartz, R.R.R. Barbosa, N. Meratnia, G. Heijenk, H. Scholten, A simple and robust dissemination protocol for VANETs, in: 2010 European Wireless Conference (EW), IEEE, 2010, pp. 214–222. doi:10.1109/EW.2010.5483419.
[10] O. Tonguz, N. Wisitpongphan, F. Bai, DV-CAST: a distributed vehicular broadcast protocol for vehicular ad hoc networks, IEEE Wireless Communications 17 (2) (2010) 47–57, doi:10.1109/MWC.2010.5450660.
[11] M. Khabbazian, Efficient broadcasting in mobile Ad Hoc networks, IEEE Transactions on Mobile Computing 8(2) (2009) 231–245, doi:10.1109/TMC.2008.94.
[12] E.M. van Eenennaam, Providing over-the-horizon awareness to driver support systems by means of multi-hop ad hoc vehicle-to-vehicle communication, Master's Thesis, University of Twente, The Netherlands, 2008.
[13] J.J. Blum, A. Eskandarian, Avoiding timeslot boundary synchronization for multihop message broadcast in vehicular networks, in: IEEE 69th Vehicular Technology Conference: VTC2009-Spring 26–29 April 2009, Hilton Diagonal Mar Hotel, Barcelona, Spain, 2009.
[14] T. Nadeem, P. Shankar, L. Iftode, A comparative study of data dissemination models for vanets, in: 3rd Annual International Conference on Mobile and Ubiquitous Systems (MOBIQUITOUS), 2006.
[15] T.D.C. Little, A. Agarwal, An information propagation scheme for VANETs, IEEE Intelligent Transportation Systems, Proceedings (2005) 155–160.
[16] O. Tonguz, N. Wisitpongphan, F. Bai, P. Mudalige, V. Sadekar, Broadcasting in VANET, Mobile Networking for Vehicular Environments (2007) 7–12.
[17] Q. Yu, G. Heijenk, Abiding geocast for warning message dissemination in vehicular ad hoc networks, in: Proceedings IEEE Vehicular Networks and Applications Workshop, 2008.
[18] A. Vahdat, D. Becker, Epidemic routing for partially connected ad hoc networks, Tech. rep., Technical Report CS-200006, Duke University, 2000.
[19] T. Spyropoulos, C.S. Psounis K. asd Raghavendra, Efficient routing in intermittently connected mobile networks: the multiple-copy case, IEEE/ACM Transactions on Networking (TON) 16 (1) (2008) 77–90.
[20] R.C. Shah, S. Roy, S. Jain, W. Brunette, I. Res, W.A. Seattle, Data mules: Modeling a three-tier architecture for sparse sensor networks, in: Proceedings of the First IEEE 2003 IEEE International Workshop on Sensor Network Protocols and Applications, 2003, pp. 30–41.
[21] P. Hui, J. Crowcroft, E. Yoneki, Bubble rap: social-based forwarding in delay tolerant networks, in: Proceedings of the 9th ACM International Symposium on Mobile ad hoc Networking and Computing, ACM, 2008, pp. 241–250.
[22] F.J. Ros, P.M. Ruiz, I. Stojmenovic, Reliable and efficient broadcasting in vehicular ad hoc networks, in: Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th, 2009, pp. 1–5. doi:10.1109/VETECS.2009.5073834.
[23] IEEE Draft Amendment to Standard for Information Technology – Telecommunications and Information exchange between systems – Local and Metropolitan networks – specific requirements - Part 11: Wireless LAN Medium Access Control, IEEE 802.11p Draft 2.0.
[24] R.S. Schwartz, M. van Eenennaam, G. Karagiannis, G. Heijenk, W. Klein Wolterink, H. Scholten, Using V2V communication to create Over-the-horizon Awareness in multiple-lane highway scenarios, in: 2010 IEEE Intelligent Vehicles Symposium, IEEE, 2010, pp. 998–1005. doi:10.1109/IVS.2010.5548073.
[25] L.Quadstone, Quadstone Paramics v5.0 Modeller user guide, Scotland, UK.
[26] D. Krajzewicz, G. Hertkorn, C. Rössel, P. Wagner, SUMO (Simulation of Urban MObility)-an open-source traffic simulation, in: Proceedings of the 4th Middle East Symposium on Simulation and Modelling (MESM20002), 2002, pp. 183–187. Available from: <http://elib.dlr.de/6661/>
[27] S.Krauß, Microscopic modeling of traffic flow: Investigation of collision free vehicle dynamics, Ph.D Thesis. Available <http://opensigle.inist.fr/handle/10068/170084>.
[28] M. Treiber, A. Hennecke, D. Helbing, Congested traffic states in empirical observations and microscopic simulations (2000) 1805–1824 (doi:10.1103/PhysRevE.62.1805) (http://arxiv.org/abs/cond-mat/0002177).
[29] M.van Eenennaam, W.K. Wolterink, G.Karagiannis, G.Heijenk, Exploring the solution space of beaconing in VANETs, in: 2009 IEEE Vehicular Networking Conference (VNC), IEEE, 2009, pp. 1–8. (doi:10.1109/VNC.2009.54163700).